Real-time Detection of Low-Rate DDoS Attacks in SDN-based Networks using Online Machine Learning Model

Abdussalam Ahmed Alashhab[1], Mohd Soperi Zahid[1], Mujaheed Abdullahi[1], and Siddikur Rahman[1]

[1]Affiliation not available

December 28, 2023

Real-time Detection of Low-Rate DDoS Attacks in SDN-based Networks using Online Machine Learning Model

Abdussalam Ahmed Alashhab [1]
Faculty of Information Technology, Alasmarya Islamic University.
Department of Computer and Information Science,
Universiti Teknologi Petronas,
Seri Iskandar, Malaysia
abdussalaam91@gmail.com

Mujaheed Abdullahi [3]
Department of Computer and Information Science,
Universiti Teknologi Petronas
Seri Iskandar, Malaysia
abdullah_18001208@utp.edu.my

Mohd Soperi Mohd Zahid [2]
Department of Computer and Information Science,
Universiti Teknologi Petronas
Seri Iskandar, Malaysia
msoperi.mzahid@utp.edu.my

Md Siddikur Rahman [4]
Department of Electrical and Electronics Engineering,
Universiti Teknologi Petronas
Seri Iskandar, Malaysia
md_22008074@utp.edu.my

***Abstract*— Software Defined Networks (SDN) provide rapid configuration, scalability, and management through a dynamic, programmable architecture that surpasses traditional network limitations. However, detecting Distributed Denial of Service (DDoS) attacks remains challenging, threatening both traditional and SDN-based networks. Machine Learning (ML) and Deep Learning (DL) technologies in conjunction with SDN have shown significant potential in effectively countering these threats. Prior studies primarily addressed high-rate DDoS attacks, neglecting low-rate DDoS attacks that resemble legitimate traffic, and often using outdated datasets. While researchers employ various offline learning algorithms to identify DDoS attacks, online learning classifiers remain underexplored. Our goal is to offer an intrusion detection model tailored to SDN networks, using the online passive-aggressive classifier. The proposed model achieves a 99.7% average detection rate for normal vs. DDoS network traffic, outperforming similar models on multiple datasets, including (CICDDoS2019, and InSDN. slow-read-DDoS), effectively detecting and mitigating DDoS attacks.**

Keywords*— *SDN; LDDoS attack; OpenFlow; Online Machine Learning; PA Classifier.

I. INTRODUCTION

Software Defined Networking (SDN) is a modern network architecture designed to surpass the limitations of traditional networks [1]. By separating the control plane responsible for routing and interfaces from the data plane handling traffic redirection, SDN offers greater flexibility and responsiveness to changing demands. Additionally, it enables network programmability, unified control capabilities, and a global view of the network topology in the controller [2, 3], making it a popular choice across various sectors. However, SDN is not immune to security vulnerabilities that can be exploited across its architectural planes. This paper focuses on Distributed Denial of Service (DDoS) attacks, a major threat to computer networks.

DDoS attacks pose a growing and complex challenge, becoming more severe with the advancement of the Internet, including the Internet of Things (IoT) and 5th generation (5G) technology [4]. These highly destructive attacks target specific network segments to disrupt normal system services. Low-rate DDoS attacks (LDDoS) have recently emerged as a distinct type, differing from traditional high-rate and volumetric DDoS attacks. LDDoS attacks send packets at a rate below the network or system capacity, aiming to exploit vulnerabilities and overwhelm resources over a more extended period. Detecting LDDoS attacks is challenging, as they generate traffic below the threshold of conventional anomaly detection methods [5].

While machine learning techniques have been used to detect LDDoS attacks in SDN-based networks, many existing approaches are designed for batch processing and lack real-time capabilities [6]. To address these issues, this paper proposes an online machine learning model employing the passive-aggressive (PA) classifier [7] for LDDoS attack detection in SDN-based networks. The proposed model processes large volumes of network traffic data in real time and updates model parameters incrementally using the PA classifier. We evaluated the model's performance on several datasets, including CICDDoS2019 [8], InSDN [9], slow-read-DDoS [10], and a custom dataset generated from simulated network traffic scenarios using Mininet [11] and the Ryu controller [12]. Our results demonstrate that the proposed model achieves high accuracy and outperforms existing methods in detecting LDDoS attacks in SDN-based networks. This paper contributes in the following ways:

- Development of an online model for LDDoS attack detection in SDN-based networks.
- The proposed model is effective in detecting LDDoS attacks while maintaining a low false positive rate, using various datasets, including custom simulated traffic scenarios.
- The proposed model is superior compared to existing methods in detecting and mitigating LDDoS attacks.

The rest of this paper is organized as follows. In Section II, we review related work on LDDoS attack detection and SDN-enabled networks. Section III provides background information on SDN, LDDoS, Online machine learning, and the PA classifier. In Section IV, we present a detailed description of the proposed methodology, including both offline and online training phases. We present and analyze the experimental results in Section V. Finally, in Section VI, we conclude the paper and discuss potential directions for future research.

TABLE I. A comparison of the related work discussed in this section.

Study	Approach	Detection Methodology	Limitations
Cheng et al. [13]	Machine learning-based	Leveraging machine learning algorithms	Ineffective in varying IoT network conditions, limited evaluation of real-world scenarios
Nadeem et al. [14]	RNN-based approach	Utilizing flow rule features and RNN	Requires a large amount of training data and computational resources, limited dataset usage, potential lack of real-world accuracy
Tang et al. [15]	Performance and Features Framework	Machine learning with OpenFlow traffic features	Limited dataset usage, no evaluation against zero-day attacks, may not capture network complexity in real-world
Proposed approach	Online machine learning-based	Flow-based and packet-based traffic data, PA classifier	N/A

II. Related Work

Low-rate Distributed Denial of Service (LDDoS) attacks have become a significant threat to network security due to their ability to evade traditional DDoS detection techniques [16]. LDDoS attacks operate at low traffic rates, which can go undetected by traditional detection methods. Limited research has been conducted on the detection of LDDoS attacks in SDN-enabled networks. Previous studies have explored various approaches, including machine learning-based methods, statistical techniques, and hybrid approaches. However, the existing literature in this specific domain is relatively scarce.

Machine learning-based approaches have shown promising results in detecting LDDoS attacks due to their ability to learn from historical traffic patterns and detect anomalies. For example, Cheng et al. [13] propose a machine learning-based approach for detecting LDDoS attacks in SDN-enabled IoT networks. The proposed method leverages machine learning algorithms to identify LDDoS attacks, which are particularly challenging to detect due to their similarity to legitimate network traffic. By utilizing SDN's programmable architecture and centralized control, the model processes large volumes of data in real time, making it suitable for IoT networks with varying traffic patterns. The experimental results demonstrate the effectiveness of the approach in accurately detecting low-rate DDoS attacks in SDN-based IoT networks. However, the proposed method is ineffective in varying IoT network conditions, such as changing traffic patterns and dynamic network topologies, on the model's performance.

Nadeem et al. [14] addressed the challenge of detecting LDDoS attacks in SDN environments. The proposed method is based on Recurrent Neural Networks (RNN) to intelligently detect LDDoS attacks. The RNN uses flow rule features for detection and is integrated into the SDN controller, and its deployment in a real-virtual network environment using the Ryu controller and Mininet demonstrates its effectiveness. However, the study used a limited dataset and evaluated the approach only on a simulated environment, which may not reflect real-world scenarios accurately.

Tang et al. [15] propose a lightweight and real-time framework called "Performance and Features" (P&F). P&F leverages machine learning to analyze traffic features extracted with OpenFlow and classifies them into two categories. It determines the effectiveness of LDoS attacks based on the performance of normal traffic under attack states (P) and locates attack sources and victims using flow features (F) based on time-frequency analysis. P&F sets corresponding mitigation schemes based on detection and locating results. Experimental results demonstrate that P&F achieves high detection rates and low false positive rates for detecting LDoS attacks. However, the study used a limited dataset and did not evaluate the effectiveness of the approach against zero-day attacks. Additionally, the approach may not perform well in real-world scenarios, as it is based on statistical features that may not capture the complexity of network traffic accurately.

Table 1 provides a comparison of related work. In contrast, our proposed approach using online machine learning utilizing PA classifier can process large amounts of data in real-time and produce an interpretable model with high accuracy, without the limitations of the other approaches.

III. Background: SDN, LDDoS, OML, and PAC

This section presents an overview of Software-Defined Networking (SDN), Low-rate DDoS attacks, and passive-aggressive classifier.

A. *Software-Defined Network (SDN)*

SDN is a revolutionary architecture addressing traditional network limitations. It separates control and data planes, enabling centralized management through an SDN controller. This offers greater flexibility, scalability, and simplified network management. Administrators can easily deploy devices from diverse vendors and dynamically adjust configurations to meet changing requirements [17]. The SDN architecture consists of three layers, as depicted in Figure 1. SDN's three layers (infrastructure, control, and application) align with the OSI model. The control plane, governed by the

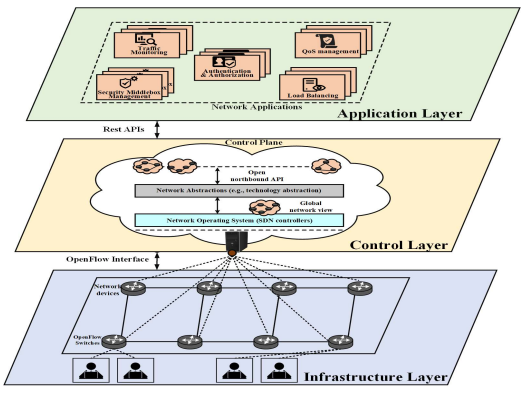


Fig. 1: Software Defined Networking Architecture.

SDN controller, makes decisions executed by the data plane across all devices. The application layer fulfils specific functions, catering to IoT requirements, and facilitating tasks like cloud storage and client-server connectivity. SDN provides a comprehensive view, enabling easy and efficient network management.

B. Low-Rate Distributed Denial Of Service Attack (LDDoS)

LDDoS is a variant of a DDoS attack employing a different method. Instead of flooding the target with large data flows, LDDoS directs a small amount of malicious flow, comprising only 20% or less of the network traffic. This low attack rate allows it to conceal within normal traffic, making detection challenging [18, 19]. To address the challenge of detecting LDDoS attacks, researchers have proposed various detection techniques. An extensive analysis of detecting LDDoS attacks in software-defined networks is presented in the study conducted in [5]. The current state-of-the-art detection methods fall into three categories: feature detection, time-domain detection, and frequency-domain detection. Feature detection creates a dataset with known LDDoS attack characteristics and evaluates ongoing flows for possible attacks. Frequency-domain methods use multifractal features and techniques like spectral analysis and wavelet transform to identify changes in the frequency domain indicating an LDDoS attack. Time-domain detection compares calculated values against a threshold using algorithms like autocorrelation to detect attack flows [20].

C. Online Machine Learning

Online machine learning is a type of machine learning in which a model is trained to learn from data that is continuously streaming into the system. In online learning, the model is presented with a sequence of data points, and it updates its predictions or actions based on the new information it receives. This process is repeated over time as the model receives more data, allowing it to adapt and improve its performance [21].

Online learning is often used in applications where data is being created continuously, such as in real-time data streams, and where it is not practical to wait until all of the data is available before starting to learn. One of the main advantages of online learning is that it can be more efficient and scalable than traditional batch learning, as the model can begin to learn and make predictions almost immediately, rather than having to wait until all of the data is available [22, 23].

As depicted in Figure 2a, batch learning, involves training a machine learning model on all available data and then storing and deploying the model as is, without further learning. This procedure can be time-consuming, particularly when dealing with large amounts of data. The model is trained and tested using the available data and then deployed. Once deployed, the model may be updated, but it will not continue learning from new data. It is important to consider the time required for learning when updating a batch learning model.

As depicted in Figure 2b, Online learning is a type of machine learning in which a model is continuously updated with small amounts of new data as it becomes available. This allows the model to continually learn and adapt to changing data patterns. Here are the key steps in online learning:

- The model is trained and deployed with a little amount of data.

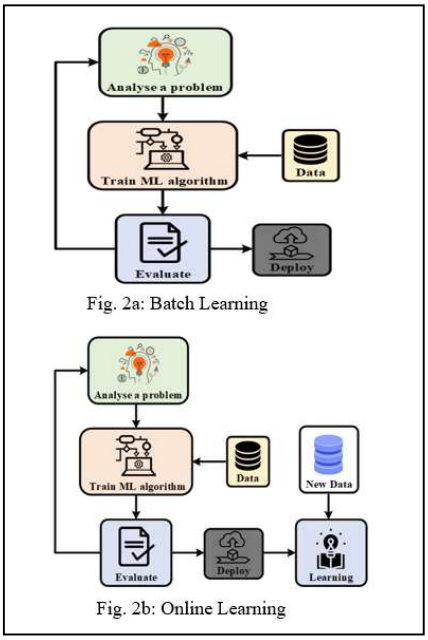


Fig. 2: Batch and Online Machin Learning

- As new data becomes available, the model updates itself with small amounts of this data, either single data points or mini-batches.
- The model continues to learn and adapt to changing data patterns even after it has been deployed.

Online learning is particularly useful in situations where the data being processed is constantly changing, such as in detecting DDoS attacks. A DDoS attack detection system needs to be able to transiently adapt to new traffic, so an online learning model that can continuously update itself is essential. It is important to consider the data that is received and how it can be used to update the model in real time.

D. Passive Aggressive (PA) Classifier

The Passive Aggressive (PA) Classifier is a machine learning algorithm used for binary classification tasks, including detecting DDoS attacks. It excels in online learning, continuously updating its model as new data arrives. The PA Classifier aggressively updates the model when it makes incorrect predictions, while still maintaining a passive learning approach [24]. It is well-suited for real-time DDoS attack detection, as it adapts to changing network patterns efficiently. The classifier uses feature vectors representing network traffic attributes and evaluates whether the input corresponds to normal traffic or a DDoS attack.

In the context of LDDoS attack detection in SDN-based networks, the PA classifier offers several advantages. It allows for real-time processing and efficient use of computational resources, making it suitable for processing large volumes of network traffic in real time.

IV. PROPOSED METHODOLOGY

This section introduces our novel approach for the online detection of LDDoS attacks in SDN-based networks. Our proposed model is based on the online Passive Aggressive (PA) classifier, enabling accurate and effective detection of LDDoS attacks.

A. Proposed Model

Our proposed online machine learning model utilizes the PA classifier to effectively identify LDDoS attacks in SDN-based networks. The model is designed to process large amounts of network traffic data in real time and update its parameters incrementally. The proposed model process flow is shown in Figure 3, involves the following steps:

1. Data Collection: We collect data from two sources. First, we use datasets like CICDDoS2019, InSDN, and slow-read-DDoS, which contain different types of DDoS attacks. These datasets include both legitimate and malicious entries to enable a comprehensive understanding of normal system behavior and the detection of known as well as novel attack patterns. Second, we create a custom dataset of network traffic data in SDN-based networks, including regular traffic and LDDoS attacks. Mininet and Ryu's controller simulates an SDN environment for data collection.
2. Data Preprocessing: To ensure model accuracy, we preprocess the collected data by removing irrelevant features and normalizing them. We perform exploratory data analysis to prepare the data for the PA classifier. Preprocessing steps include handling missing data and columns, transforming raw data into refined datasets, and establishing consistent feature types across all datasets.
3. Data Training: Our online learning model comprises two stages: offline training and online learning. In the offline training stage, the primary datasets are used to train the model and create a primary database from preprocessed data. The online learning stage continuously trains the model on new data or traffic, one sample at a time. The PA optimizer incrementally updates the model parameters, enabling real-time recognition of new data.
4. Model Evaluation: We evaluate the proposed model using various datasets, including CICDDoS2019, InSDN, slow-read-DDoS, and our generated dataset. Metrics such as accuracy, loss rate, precision, recall, and F1-score are used to assess model performance. The evaluation process informs practical decisions based on the model's outcomes.
5. Deployment: The proposed model is integrated into an SDN-based network to enable real-time LDDoS attack detection. This deployment involves configuring the model to continuously monitor network traffic for signs of LDDoS attacks. As depicted in Figure 3, the model undergoes testing and fine-tuning to ensure its effectiveness in a live environment. Ongoing monitoring and regular updates help maintain its efficiency in detecting LDDoS attacks.
6. Incremental Training: To optimize computational resources and keep the model up to date with emerging LDDoS attack patterns, we implement incremental training. This approach allows us to adapt the model's parameters using new data collected during the data collection phase. By training incrementally with sequential data instances, the model evolves in real time, ensuring the swift and accurate detection of LDDoS attacks as they occur.

B. Online Training of the Proposed Model

After completing the offline training phase, which involves training the model on the primary datasets and creating a primary database using pre-processed data, the online machine learning model takes over. Figure 4 illustrates

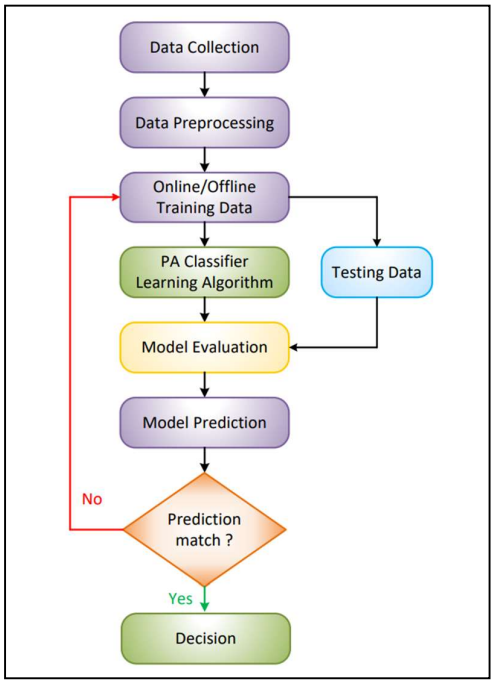


Fig. 3: Proposed Model Process Flow.

the pseudocode for the online training process of the PA classifier. In the pseudocode, data represents the input feature vectors, labels represent the corresponding true labels (1 for positive class, -1 for negative class), regularization_parameter controls the aggressiveness of updates, and max_iterations determines the number of passes through the entire dataset.

During the online training process, the PA classifier updates its weight vector **w** and bias term **b** incrementally for each instance in the dataset. If an instance is misclassified, the model performs an aggressive update to correct the mistake. The learning rate **alpha** is calculated based on the loss and regularization parameters, ensuring that the model adjusts the weights and biases appropriately. The online training process allows the PA classifier to adapt to new data and continuously improve its performance as new instances are fed into the model.

```
Function online_PA_training(data, labels,
regularization_parameter, max_iterations):
    Initialize weight vector w with zeros or small random values
    Initialize bias term b to 0
    for iteration in range(max_iterations):
        for i in range(len(data)):
            instance = data[i]
            true_label = labels[i]
            prediction = sign(w • instance + b)
            loss = max(0, 1 - true_label * (w • instance + b))
            if loss > 0:
                alpha = loss / (||instance||^2 + 1 / (2 *
regularization_parameter))
                w = w + alpha * true_label * instance
                b = b + alpha * true_label
    return w, b
```

Fig. 4: Pseudo code of PA online training process.

V. RESULTS AND DISCUSSION

This section, describs the experimental setup and results obtained from our proposed online machine learning model using a PA classifier to detect LDDoS attacks in SDN-based networks.

A. Experimental Setup

We conducted the experiments on the Mininet simulator and created a Fat-tree network topology using the Python API of Mininet as shown in Figure 5. The topology consisted of one Ryu controller, ten OpenFlow switches, and eighty hosts. The bandwidths were adjusted to 10Mbps and 100Mbps, representing Ethernet and Fast Ethernet connections, individually. Normal traffic was generated using Ping and LDDoS attacks using Scapy.

We used Python and scikit-learn to implement the proposed model. The AP algorithm was used for the online learning part of the model. The training dataset consisted of 100,000 samples, with 70% used for training and 30% used for testing. Normalization was applied using the min-max data normalization technique, as depicted in equation (1).

$$\mathbf{x}_i' = \frac{\mathbf{x}_i - min\ \mathbf{x}_i}{max\ \mathbf{x}_i - min\ \mathbf{x}_i} \qquad (1)$$

In evaluating the model's performance, we use the following performance metrics:

Accuracy, which measures the proportion of correctly classified instances, serves as the evaluation metric in this research. The performance of the classification model was assessed based on various parameters, with accuracy being the focus for measuring the single-class accuracy of the model. The accuracy of the proposed online model was determined using a specific equation as depicted in equation (2).

$$\text{Accuracy} = \frac{(\text{tp} + \text{tn})}{(\text{tp} + \text{fp} + \text{tn} + \text{fn})} \qquad (2)$$

The symbols tp, tn, fp, and fn represent true positive, true negative, false positive, and false negative, respectively.

Precision is defined as the proportion of true positives out of all predicted positives. For the proposed online model, the calculation of precision was performed using equation (3).

$$\text{Precision} = \frac{\text{tp}}{(\text{tp} + \text{fp})}, \qquad (3)$$

Recall, also known as sensitivity or true positive rate, measures the proportion of true positives out of all actual positives. In the context of the proposed online model, the calculation of recall was determined using equation (4).

$$\text{Recall} = \frac{\text{tp}}{(\text{tp} + \text{fn})}, \qquad (4)$$

The F1-score, a balanced measure of performance, is calculated as the harmonic mean of precision and recall. In the case of the proposed online model, the F1-score was obtained using equation (5).

$$\text{F-measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}, \qquad (5)$$

Loss Rate, This metric signifies the fraction of misclassified instances. We calculated the loss rate using equation (6).

$$\text{LossRate} = \frac{(\text{Misclassified Instances})}{(\text{Total Instances})} \qquad (6)$$

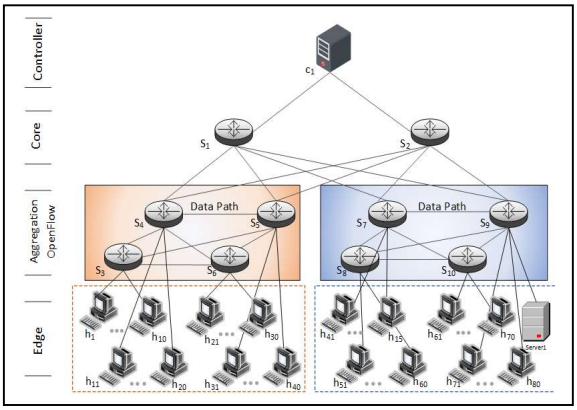


Fig. 5: Network Topology used in the Experiment.

B. Results

Table II describes the performance metrics of the proposed model on the training and validation data for four datasets: CICDDoS2019, InSDN, slow-read-DDoS, and our custom dataset. The metrics assessed include accuracy, loss rate, precision, recall, and F1-score. The performance results of the method employed in this research are visualized in Figure 6.

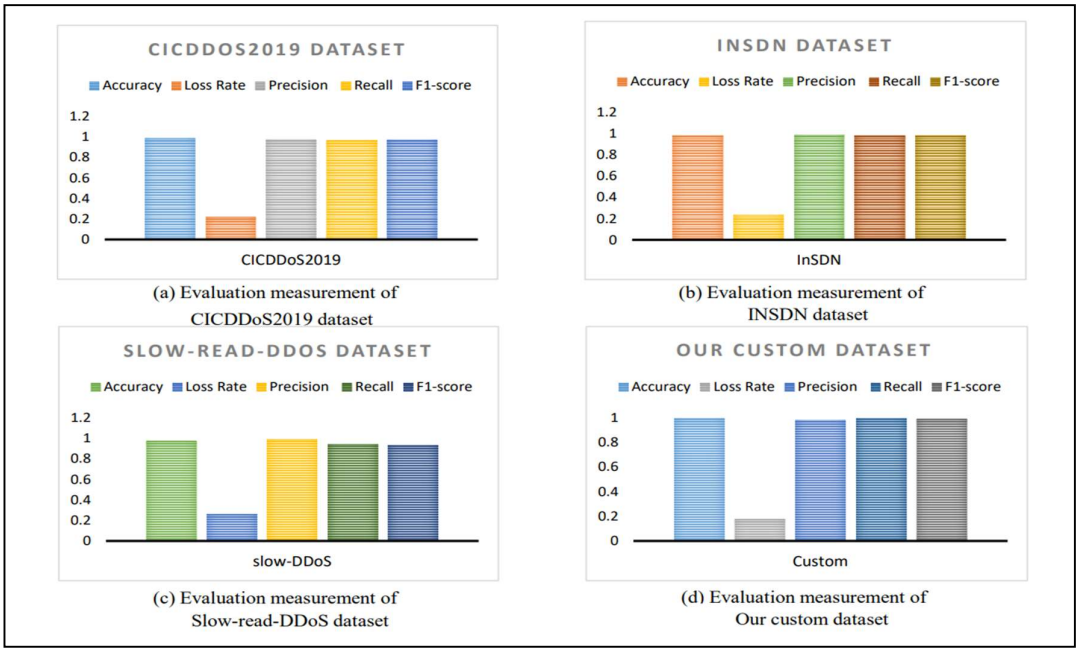
The results of the proposed online model on the training data of CICIDS2019 achieved an accuracy of 99% with a low loss rate of 0.25%. The model also achieved high precision (0.9746), recall (0.9657), and F1-score (0.9691) as obtained in Figure 6a. Similarly, on the training data of InSDN, the model achieved high accuracy (98%) and low loss rate (0.325%), with precision, recall, and F1-score of 0.9866, 0.9847, and 0.9791, respectively shown in figure 6b.

On the training data of the slow-read-DDoS dataset, the proposed online EBM model achieved high accuracy (97%) and low loss rate (0.22%). The model also achieved high precision (0.9895), recall (0.9423), and F1-score (0.9291), respectively shown in Figure 6c. The results on the training data of our custom dataset were even better, with high accuracy (99%), low loss rate (0.12%), precision (0.9795), recall (0.9923), and F1-score (0.9891), respectively shown in figure 6d.

The validation results of the proposed model were consistent with the training results, indicating that the model generalizes well to new data. The model's performance was also competitive compared to other state-of-the-art models in the literature. Overall, the experimental results show that our proposed online machine learning model is effective in detecting LDDoS attacks in SDN-based networks with high accuracy and a low false positive rate.

TABLE II performance metrics of the proposed model.

Dataset	Accur -acy	Loss Rate	Precis- ion	Recall	F1-score
CICDDoS2019,	0.988	0.218	0.9746	0.9657	0.9691
InSDN	0.984	0.233	0.9866	0.9847	0.9791
slow-DDoS	0.972	0.262	0.9895	0.9423	0.9291
Custom dataset	0.997	0.174	0.9795	0.9923	0.9891



(a) Evaluation measurement of CICDDoS2019 dataset

(b) Evaluation measurement of INSDN dataset

(c) Evaluation measurement of Slow-read-DDoS dataset

(d) Evaluation measurement of Our custom dataset

Fig. 6: Performance of the Proposed Model.

We compared our proposed model's performance with existing methods in the literature, as shown in Table III. Our proposed model outperformed the existing methods in terms of accuracy, precision, recall, and F1-score.

C. *Discussion*

The proposed model achieved high accuracy, low loss rate, and high precision, recall, and F1-score on the training data. Moreover, the model's ability to continuously learn from incoming data and adapt to varying network conditions makes it more suitable for use in dynamic network environments compared to traditional machine learning-based methods that require periodic retraining.

Compared to other machine learning-based methods, such as decision tree, KNN, and SVM, the proposed model has better performance in terms of accuracy and interpretability. Furthermore, the proposed model can be trained with PA, which is a widely used and efficient optimization algorithm suitable for large-scale datasets.

One potential limitation of the proposed model is that it may require more computational resources for training and inference compared to simpler methods, such as decision tree or KNN. However, this is mitigated by the fact that the proposed model can operate in an online learning mode, allowing it to adapt to changing network conditions without requiring periodic retraining.

VI. Conclusion

In this paper, we proposed an online machine learning model using a PA classifier to detect LDDoS attacks in SDN-based networks. The proposed model achieved high accuracy and a low loss rate on the training data. However, there are some limitations to our proposed model. Firstly, the model was tested in a simulated environment using Mininet, which may not fully reflect the complexities of a real-world SDN-based network. Secondly, the proposed model was only evaluated on LDDoS attacks, and not on other types of attacks.

In future research, we plan to evaluate the proposed model in a real-world environment and test its performance on a wider range of attacks. We also aim to explore the use of other online machine learning algorithms and investigate the potential benefits of using a combination of multiple algorithms for improved accuracy and efficiency. Additionally, we will investigate the integration of the proposed model with existing security frameworks in SDN-based networks.

REFERENCES

[1] D. Kreutz, F. M. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE,* vol. 103, no. 1, pp. 14-76, 2014.

TABLE III COMPARISON OF PROPOSED MODEL PERFORMANCE WITH THE RELATED LITERATURE CONTRIBUTIONS

References	Year	Layer Location	Classifier/ Method	Dataset	SDN Controller	Detection Results
Cheng et al. [13]	2020	Control Layer Data layer	SVM, NB, KNN, RF	Custom Dataset	Floodlight	97%
Nadeem et al. [14]	2022	Data layer	RNN	Custom Dataset	Ryu	98.5%
Tang et al. [15]	2021	Control Layer and data layer	RF, MLP	Custom Dataset	ONOS	98%
Our Work	2023	Control Layer and Data Layer	Online PA classifier	CICDDoS2019, InSDN, slow-DDoS, Custom dataset	Ryu	99.7%

[2] L. Zhu *et al.*, "SDN controllers: A comprehensive analysis and performance evaluation study," *ACM Computing Surveys (CSUR),* vol. 53, no. 6, pp. 1-40, 2020.

[3] A. A. Alashhab, M. S. M. Zahid, A. A. Barka, and A. M. Albaboh, "Experimenting and evaluating the impact of DoS attacks on different SDN controllers," in *2021 IEEE 1st International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering MI-STA*, 2021: IEEE, pp. 722-727.

[4] F. S. Dantas Silva, E. Silva, E. P. Neto, M. Lemos, A. J. Venancio Neto, and F. Esposito, "A taxonomy of DDoS attack mitigation approaches featured by SDN technologies in IoT scenarios," *Sensors,* vol. 20, no. 11, p. 3078, 2020.

[5] A. A. Alashhab, M. S. M. Zahid, M. A. Azim, M. Y. Daha, B. Isyaku, and S. Ali, "A Survey of Low Rate DDoS Detection Techniques Based on Machine Learning in Software-Defined Networks," *Symmetry,* vol. 14, no. 8, p. 1563, 2022.

[6] N. Tantalaki, S. Souravlas, and M. Roumeliotis, "A review on big data real-time stream processing and its scheduling techniques," *International Journal of Parallel, Emergent and Distributed Systems,* vol. 35, no. 5, pp. 571-601, 2020.

[7] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive aggressive algorithms," 2006.

[8] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy," in *2019 International Carnahan Conference on Security Technology (ICCST)*, 2019: IEEE, pp. 1-8.

[9] M. S. Elsayed, N.-A. Le-Khac, and A. D. Jurcut, "InSDN: A novel SDN intrusion dataset," *Ieee Access,* vol. 8, pp. 165263-165284, 2020.

[10] N. M. Yungaicela-Naula, C. Vargas-Rosales, J. A. Pérez-Díaz, and D. F. Carrera, "A flexible SDN-based framework for slow-rate DDoS attack mitigation by using deep reinforcement learning," *Journal of Network and Computer Applications,* vol. 205, p. 103444, 2022.

[11] N. Handigol, B. Heller, V. Jeyakumar, B. Lantz, and N. McKeown, "Reproducible network experiments using container-based emulation," in *Proceedings of the 8th international conference on Emerging networking experiments and technologies*, 2012, pp. 253-264.

[12] S. Asadollahi, B. Goswami, and M. Sameer, "Ryu controller's scalability experiment on software defined networks," in *2018 IEEE international conference on current trends in advanced computing (ICCTAC)*, 2018: IEEE, pp. 1-5.

[13] H. Cheng, J. Liu, T. Xu, B. Ren, J. Mao, and W. Zhang, "Machine learning based low-rate DDoS attack detection for SDN enabled IoT networks," *International Journal of Sensor Networks,* vol. 34, no. 1, pp. 56-69, 2020.

[14] M. W. Nadeem, H. G. Goh, Y. Aun, and V. Ponnusamy, "A Recurrent Neural Network based Method for Low-Rate DDoS Attack Detection in SDN," in *2022 3rd International Conference on Artificial Intelligence and Data Sciences (AiDAS)*, 2022: IEEE, pp. 13-18.

[15] D. Tang, Y. Yan, S. Zhang, J. Chen, and Z. Qin, "Performance and features: Mitigating the low-rate TCP-targeted DoS attack via SDN," *IEEE Journal on Selected Areas in Communications,* vol. 40, no. 1, pp. 428-444, 2021.

[16] M. Chen, J. Chen, X. Wei, and B. Chen, "Is low-rate distributed denial of service a great threat to the Internet?," *IET Information Security,* vol. 15, no. 5, pp. 351-363, 2021.

[17] D. Kumar and J. Thakur, "Handling Security Issues in Software-defined Networks (SDNs) Using Machine Learning," in *Computational Vision and Bio-Inspired Computing*: Springer, 2022, pp. 263-277.

[18] A. Shevtekar, K. Anantharam, and N. Ansari, "Low rate TCP denial-of-service attack detection at edge routers," *IEEE Communications Letters,* vol. 9, no. 4, pp. 363-365, 2005.

[19] A. A. Alashhab, M. S. M. Zahid, A. Muneer, and M. Abdullahi, "Low-rate DDoS attack Detection using Deep Learning for SDN-enabled IoT Networks," *International Journal of Advanced Computer Science and Applications,* vol. 13, no. 11, 2022.

[20] C. Zhang, J. Yin, Z. Cai, and W. Chen, "RRED: robust RED algorithm to counter low-rate denial-of-service attacks," *IEEE Communications Letters,* vol. 14, no. 5, pp. 489-491, 2010.

[21] Ó. Fontenla-Romero, B. Guijarro-Berdiñas, D. Martinez-Rego, B. Pérez-Sánchez, and D. Peteiro-Barral, "Online machine learning," in *Efficiency and Scalability Methods for Computational Intellect*: IGI global, 2013, pp. 27-54.

[22] C. S. Lee and A. Y. Lee, "Clinical applications of continual learning machine learning," *The Lancet Digital Health,* vol. 2, no. 6, pp. e279-e281, 2020.

[23] A. A. A. M. S. M. Z. M. A. S. Alashhab, "Online Machine Learning Approach to Detect and Mitigate Low-Rate DDoS Attacks in SDN-Based Networks," presented at the 2023 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), Kota Kinabalu, Malaysia, 2023.

[24] S. Ifzarne, H. Tabbaa, I. Hafidi, and N. Lamghari, "Anomaly detection using machine learning techniques in wireless sensor networks," in *Journal of Physics: Conference Series*, 2021, vol. 1743, no. 1: IOP Publishing, p. 012021.